Real-time end-to-end guarantees for the EF classwith and without traffic shaping

Steven Martin, Pascale Minet, Laurent George

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Real-time end-to-end guarantees for the EF class with and without traffic shaping

Steven MARTIN — Pascale MINET — Laurent GEORGE

N° 4980

October 2003

THÈME 1

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--4980--FR+ENG

Real-time end-to-end guarantees for the EF class with and without traffic shaping

Steven MARTIN* , Pascale MINET† , Laurent GEORGE‡

Thème 1 — Réseaux et systèmes
Projet HIPERCOM

Rapport de recherche n° 4980 — October 2003 — 34 pages

Abstract: In this paper, we are interested in providing deterministic end-to-end guarantees to the Expedited Forwarding (EF) class of the Differentiated Services (DiffServ) model. We focus on two Quality of Service (QoS) parameters: the end-to-end response time and the end-to-end jitter. As packets of any flow can experience variable network delays and sojourn times on each visited node, the inter-arrival times can be shorter than those on the source node and burst arrivals are possible. This flow distortion increases with the number of visited nodes. To cope with this distortion, traffic shaping has been introduced. We focus more particularly on two techniques of traffic shaping: jitter cancellation and token bucket. We then study the influence of traffic shaping on these two QoS parameters, independently of the scheduling policy for the EF class. In this paper, we show how to compute the worst case end-to-end response time and jitter of any flow in the EF class with and without traffic shaping, assuming that the EF class has the highest priority and packets in this class are served FIFO. We then determine when each one of the three techniques (no traffic shaping, jitter cancellation and token bucket) is the most appropriate.

Key-words: deterministic QoS guarantee, DiffServ, EF class, traffic shaping, real-time scheduling, jitter cancellation, tocken bucket.

* Ecole Centrale d'Electronique, 53 rue de Grenelle, 75007 Paris, *steven.martin@ece.fr*
† INRIA Rocquencourt, 78153 Le Chesnay Cedex, *pascale.minet@inria.fr*
‡ Université Paris 12, LIIA, 120 rue Paul Armangot, 94400 Vitry, *george@univ-paris12.fr*

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 —Télécopie : +33 1 39 63 53 30

Garanties temps-réel de bout-en-bout pour la classe EF avec et sans mise en forme du trafic

Résumé : Nous nous intéressons à la garantie déterministe d'une qualité de service (QoS) de bout-en-bout pour la classe *Expedited Forwarding* (EF) du modèle *Differentiated Services* (DiffServ). Nous étudions plus particulièrement deux paramètres de QoS : le temps de réponse de bout-en-bout et la gigue de bout-en-bout. Les paquets subissent des délais réseau variables et des durées de séjour variables dans chacun des nœuds visités. Les délais d'inter-arrivée des paquets dans les nœuds peuvent donc devenir plus petites que ceux connus dans le nœud source. Les nœuds voient alors arriver des rafales de paquets d'un même flux. Cette distorsion du trafic s'amplifie de nœud en nœud. Pour pallier ce problème de distorsion, des techniques de remise en forme du trafic ont été introduites. Nous nous intéressons dans cet article à deux d'entre elles : l'annulation de gigue et le seau à jetons. Nous étudions l'influence d'une telle remise en forme sur les deux paramètres de QoS considérés, indépendamment de la politique d'ordonnancement choisie au sein de la classe EF. Nous calculons ensuite le temps de réponse et la gigue pire cas de bout-en-bout obtenus avec et sans remise en forme du trafic, lorsque les paquets de la classe EF sont ordonnancés FIFO, cette classe étant traitée prioritairement à toute autre classe. Nous déterminons alors les cas où l'une des trois techniques étudiées (aucune remise en forme du trafic, remise en forme par annulation de gigue et remise en forme par seau à jetons) est préférable.

Mots-clés : garantie déterministe de qualité de service, DiffServ, classe EF, remise en forme du trafic, ordonnancement temps réel, annulation de gigue, seau-à-jetons.

Contents

1 Introduction

The all IP convergence leads networks to support applications with real-time constraints. Examples of such applications are voice and video flows, remote control surgery and industrial process control. These applications make the best-effort service model inadequate. Indeed, this model cannot provide Quality of Service (QoS) guarantees, in terms of bandwidth, delay, jitter or packet loss.

To answer the end-to-end QoS requirements in networks, we focus on the DiffServ architecture, that has a good scalability. This architecture is based on traffic aggregation in a limited number of classes. In particular, the *Expedited Forwarding* (EF) class has been proposed for applications requiring low end-to-end packet delay, low delay jitter and low packet loss (e.g. voice and video applications that are delay and jitter sensitive), the EF class being the highest priority class. In this paper, we propose a solution to guarantee deterministic end-to-end response times and jitters to all the EF flows in a DiffServ domain. Indeed, beyond the qualitative definition of the offered QoS, no deterministic end-to-end guarantees have been proved for the EF class.

Due to varying network delays and sojourn times in each visited node, the distortion of any flow belonging to the EF class increases with the number of visited nodes. To avoid or limit such a phenomenon, traffic shaping has been introduced. In this paper, we focus on two shaping techniques: jitter cancellation and token bucket. We study the influence of these techniques on two QoS parameters, that is the worst end-to-end response time and jitter of any EF flow. Then, we assume that the scheduling of the EF class is *First In First Out* (FIFO) and we evaluate these two QoS parameters with and without traffic shaping.

The rest of the paper is organized as follows. In Section 2, we define the problem and the different models. Section 3 briefly discusses related work. We then introduce in Section 4 the notations used throughout the paper and define the constituent parts of the end-to-end response time and jitter of any EF flow. We present in Section 5 the three studied techniques (no shaping, shaping by jitter cancellation and shaping by token bucket) and determine their effect on the end-to-end response time and jitter. Results of this section are established whatever the scheduling policy for the EF class. In Section 6, we show how to compute upper bounds on the end-to-end response time and the end-to-end delay jitter of any flow belonging to the EF class, based on a worst case analysis of FIFO scheduling, with and without traffic shaping. In Section 7, we deduce from this analysis when each technique is the most appropriate. Finally, our results are illustrated by an example in Section 8.

Moreover, all the properties given in this paper are proved in the annex.

2 The problem

We investigate the problem of providing deterministic QoS guarantees, in terms of end-to-end response time and delay jitter, to any flow belonging to the *Expedited Forwarding* class in a DiffServ domain. The end-to-end response time and delay jitter of an EF flow are defined between the ingress node and the egress node of the flow in the domain considered.

In this paper, we analyze the possibility of fulfilling the QoS requirements of any flow belonging to the EF class with and without traffic shaping. We want to provide bounds on the end-to-end response time and jitter of any EF flow. As we make no particular assumption concerning the arrival times of packets in the domain, the feasibility of a set of EF flows is equivalent to meet both constraints, whatever the arrival times of the packets in the domain.

Moreover, we assume the following models.

2.1 DiffServ Model

In the DiffServ architecture [BBC98], traffic is distributed over a small number of classes. Packets carry the code of their class. This code is then used in each DiffServ-compliant router to select predefined packet handling functions (in terms of queuing, scheduling and buffer acceptance), called *Per-Hop Behavior* (PHB). Thus, the network is divided in two parts: the nodes at the boundary of the network (ingress and egress routers), that perform complex treatments (packet classification and traffic conditioning) and the nodes in the core network (core routers), that forward packets according to their class code.

DiffServ provides coarser levels of service than IntServ due to flow aggregation, but is implementable in large networks, simplifies the construction of multi-domain services, and is well adapted for tariffing by class of service.

Several per-hop behaviors have been defined:

- the *Expedited Forwarding* PHB [JNP99]. Traffic belonging to the EF service is delivered with very low latency and drop probability, up to a negotiated rate. This service can be used for instance by IP telephony.
- the *Assured Forwarding* (AF) PHB group [HBW99]. Four classes, providing more or less resources in terms of bandwidth and buffers, are defined in the AF service. Each class manages three different drop priorities representing the relative importance of a packet in the class.
- the *Best-Effort Forwarding* PHB is the default one.

2.2 Network model

The network considered is a DiffServ domain (all routers are DiffServ-compliant). Moreover, links interconnecting routers are supposed to be FIFO and the network delay between two nodes (including the propagation delay and the medium access time) is bounded by P_{min} and P_{max}. We assume that the path followed by a flow is fixed. This can be ensured using *source routing* or MPLS [RVC01], [LFW00].

In this paper, we consider neither network failures nor packet losses.

2.3 Node model

We consider that a DiffServ-compliant router implements Best-Effort, AF and EF classes. On any node, the EF handler optionally includes a shaper per flow. Any EF packet arriving in such a node first enters the node shaper (if it exists) and then enters the node scheduler (see Figure 1). In the following, we consider three cases:

- no shaper;
- shapers applying jitter cancellation;
- shapers based on token bucket.

When a packet enters the node scheduler, it is scheduled with the other packets of its class waiting for processing. In Section 6, we will consider that the scheduling of the EF class is FIFO. Moreover, we assume that packet scheduling is non-preemptive. Therefore, the node scheduler waits for the completion of the current packet transmission (if any) before selecting the next packet.

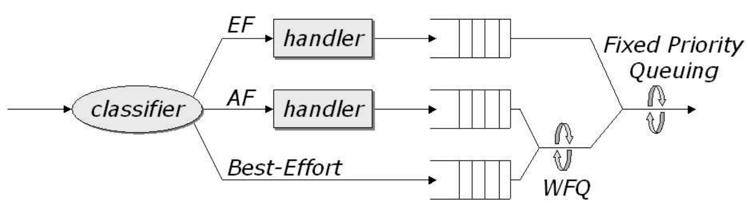

Figure 1: DiffServ-compliant router

As illustrated by Figure 1, the EF class is scheduled with a *Fixed Priority Queuing* scheme with regard to the other classes. Thus, the EF class is served as long as it is not empty. Packets in the Best-Effort and AF classes are served according to WFQ (*Weighted Fair Queuing*) [PG94]. In this way, EF traffic will obtain low delay thanks to *Fixed Priority Queuing* and AF traffic will receive a higher bandwidth fraction than Best-Effort thanks to WFQ.

2.4 Traffic model

We focus on the flows belonging to the EF class. We consider a set $\tau = \{\tau_1, ..., \tau_n\}$ of n sporadic flows belonging to the EF class. Each flow τ_i follows a sequence of nodes whose first node in the DiffServ domain is an ingress node. In the following, we call line the ordered set of nodes visited by the flow to reach a destination. Moreover, we assume that the EF flows enter the DiffServ domain without jitter. A sporadic flow τ_i is then defined by:

- T_i, the minimum inter-arrival time (abusively called period) between two successive packets of τ_i;
- L_i, the maximum length of any packet of τ_i;

This characterization is well adapted to real-time applications for which it is easy to specify those parameters. Moreover, we denote:

- C_i^h, the processing time of any packet of τ_i on node h. This quantity can be deduced from the length of the packet, the situation of node h (edge or core router) and the processing capacity of this node;
- $\tau(m)$, the index number of the EF flow which packet m belongs to;
- B^h, the maximum processing time at node h of any packet of flow not belonging to the EF class.

3 Related Work

In this section, we first present works related to the *Expedited Forwarding* class. Then, we examine two different shaping techniques to avoid traffic distortion. Finally, we scan the existing approaches to obtain deterministic end-to-end response time guarantees in a multi-hop network.

3.1 Expedited Forwarding class

The definition of the EF PHB as given in [JNP99] can be used to predict qualitative end-to-end delay guarantees. Beyond this definition, end-to-end guarantees are crucial for delay and jitter sensitive applications. [BBC01] shows that the worst case delay jitter for the EF traffic can be large in case of large networks.

The use of the FIFO scheduling algorithm for priority traffic in a network based on traffic aggregation (e.g. all flows in the EF class share a single FIFO queue) has been discussed in [CLB00]. Nevertheless, the found delay bound is valid only for reasonably small EF traffic utilization. In [GCL01], a hybrid admission control scheme has been proposed, based on traffic shaping at border routers, to provide QoS guarantees in a DiffServ environment. The decision of the admission control is based on measurements realized to estimate resource allocation, leading to a higher efficiency.

3.2 Solutions for traffic distortion

As we will see in Section 5.1, traffic distortion resulting from varying network delays and varying sojourn times in the visited nodes increases with the number of visited nodes and the number of crossed flows. To cope with the distortion, traffic shaping has been introduced. We can distinguish two techniques:

- *The jitter cancellation technique* consists in cancelling on each node the jitter of a flow before it is considered by the node scheduler [GKM01]: a flow packet is hold until its latest possible arrival time. Hence a flow packet arrives at node $h+1$ with a jitter depending only on the jitter introduced by the previous node h and the link between them. As soon as this jitter is cancelled, this packet is seen by the scheduler of node $h+1$. The worst case end-to-end response time is obtained by adding the worst case response time, without jitter (as cancelled) on every node, as we will see in Section 5.2.1. A technique derived from jitter cancellation is the *constrained jitter technique*, consisting in checking that the jitter of a flow remains bounded by a maximum acceptable value before the flow is considered by the node scheduler. If not, the jitter is reduced to the maximum acceptable value J by holding a packet until its latest possible arrival time minus J.

- *The token bucket technique* consists in cancelling on each node the bursts of a flow before it is considered by the node scheduler [PG94], [GGP96], [CS98]. The token bucket can model a flow or all flows of a DiffServ class. In the first case, it requires to maintain per flow information on every visited node. In the second case, the choice of good values for the token bucket parameters is complex when flows have different characteristics. With this model, arises the problem of fixing the good values of these parameters for a given application. As shown in [GGP96] and [SCG00], the end-to-end response times strongly depend on the choice of the token bucket parameters. Furthermore, the token bucket parameters can be optimized for a given configuration, only valid at a given time. If the configuration evolves, the parameters of the token bucket should be recomputed on every node to remain optimal. This is not generally done. Moreover, in [GGP96], it is shown that no benefit is derived from having non-identical shapers for a flow at each visited node. That is why, in the paper, all shapers for a flow have the same parameters. The end-to-end delay bound for a flow τ_i is given by the sum of the maximum shaper delay for τ_i and the sum of the maximum local scheduler delay bound for τ_i on each visited node. The maximum shaper delay is incurred only once and does not depend on the number of visited nodes.

[SC00] deals with the choice of the shaper delay and concludes that when the number of visited nodes is large, smoothing the bursts can be beneficial, but when this number is below a critical value, smoothing is detrimental to network performance. We reach a similar conclusion in Section 7.

3.3 End-to-end response time computation

To determine the maximum end-to-end response time, several approaches can be used: a stochastic or a deterministic one. A stochastic approach consists in determining the mean behavior of a network, leading to mean, statistical or probabilistic end-to-end response times ([SCG01], [VLB02]). A deterministic approach is based on a worst case analysis of the network behavior, leading to worst case end-to-end response times ([CS98], [GMM01]).

In this paper, we are interested in the deterministic approach as we want to provide a deterministic guarantee of worst case end-to-end response time and jitter for any EF flow in the network. In this context, two different approaches can be used to determine the worst case end-to-end delay: the holistic approach and the trajectory approach.

- *The holistic approach* [TC94] considers the worst case scenario on each node visited by a flow, accounting for the maximum possible release jitter introduced by the previous visited nodes. If no jitter control is done, the jitter will increase throughout the visited nodes. In this case, the minimum and maximum response times on a node h induce a maximum release jitter on the next visited node $h+1$ that leads to a worst case response time and then a maximum release jitter on the following node and so on. The holistic approach can be pessimistic as it considers worst case scenarios on every node possibly leading to impossible scenarios.

- *The trajectory approach* [LBT97] consists in examining the scheduling produced by all the visited nodes of a flow. In this approach, only possible scenarios are examined. For instance, the fluid model (see [PG94] for GPS) is relevant to the trajectory approach. This approach produces the best results as no impossible scenario is considered but is somewhat more complex to use. This approach can also be used in conjunction with a jitter control (see [GGP96] for EDF, and [PG94] for GPS).

In this paper, we adopt the trajectory approach in a DiffServ domain to determine the maximum end-to-end response time and jitter of any EF flow.

4 Notations and definitions

4.1 Notations

In the following, we focus on any EF flow τ_i following a line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes numbered from 1 to q. For any EF flow τ_j following a line $\mathcal{L}'$, with $\mathcal{L}' \bigcap \mathcal{L} \neq \emptyset$, we define $first_j$ as the first node visited by τ_j in line $\mathcal{L}$. Moreover, $slow_j$ denotes the slowest node of line $\mathcal{L}$ visited by τ_j. Thus, for any node $h \in \mathcal{L}$ visited by any EF flow τ_j, we have $C_j^h \leq C_j^{slow_j}$. Finally, we consider that on any node h, the processing time of an EF packet is less than or equal to $C_{max}^h = \max_{j=1..n}\{C_j^h\}$.

In the rest of this paper, we use the following notations, illustrated by Figure 3.

$shap(h)$	shaper of node h;
$sched(h)$	scheduler of node h;
e	element of the network such as a node, a shaper, a scheduler or a line;
a_m^e	arrival time of packet m in element e;
d_m^e	departure time of packet m from element e;
$P_m^{h,h+1}$	network delay experienced by packet m between nodes h and $h+1$;
R_m^e	response time of packet m in element e;
S_m^e	delay experienced by packet m from its arrival time in the domain to reach e;
Jin_i^e	worst case jitter of flow τ_i when entering element e;
$Jout_i^e$	worst case jitter of flow τ_i when leaving element e.

Moreover, we denote $Rmin_i^e$ (respectively $Rmax_i^e$) the minimum (respectively the maximum) response time experienced by packets of flow τ_i in element e. Thus, we have for any packet m of τ_i: $Rmin_i^e \leq R_m^e \leq Rmax_i^e$. In the same way, we denote $Smin_i^e$ (respectively $Smax_i^e$) the minimum (respectively the maximum) delay experienced by packets of flow τ_i from their arrival times in the domain to reach element e. Thus, we have for any packet m of τ_i: $Smin_i^e \leq S_m^e \leq Smax_i^e$. Finally, as assumed in Section 2.2, we have: $\forall m$ of τ_i, $Pmin \leq P_m^{h,h+1} \leq Pmax$.

4.2 Definitions

4.2.1 End-to-end response time

Let us consider any EF flow τ_i following a line $\mathcal{L}$ consisting of q nodes numbered 1 to q. The end-to-end response time of any packet m of τ_i depends on its sojourn times on each visited node and network delays. The sojourn time of packet m on any visited node consists of two parts: the waiting time in the shaper and the waiting time in the scheduler. Hence, the end-to-end response time of packet m can be divided into: (i) the delay incurred by m in all the visited shapers, (ii) the delay incurred by m in all the visited schedulers and (iii) the network delay incurred by m. Thus, we have:

$$R_m^{\mathcal{L}} = \sum_{h=1}^{q} R_m^{shap(h)} + \sum_{h=1}^{q} R_m^{sched(h)} + \sum_{h=1}^{q-1} P_m^{h,h+1}.$$

Maximizing independently each of these terms generally leads to a pessimistic upper bound. We can obtain better results in adopting the trajectory approach, as we will see in Section 6.

As detailed in Section 2.3, packet scheduling is non-preemptive. Hence, whatever the scheduling algorithm in the EF class and despite the highest priority of this class, a packet from another class (i.e. Best-Effort or AF class) can interfere with EF flows processing due to non-preemption. Indeed, if any EF packet m enters the scheduler of any node h while a packet m' not belonging to the EF class is being processed on h, m has to wait until m' completion.

It is important to notice that the non-preemptive effect is not limited to this waiting time. For example, EF packets may arrive in the node scheduler while m is waiting for processing. If they have a priority higher than packet m, then they will be processed before m. But these packets would not be necessarily processed before m if m' did not exist (see Figure 2). The resulting delay incurred by m is indirectly due to the non-preemption.

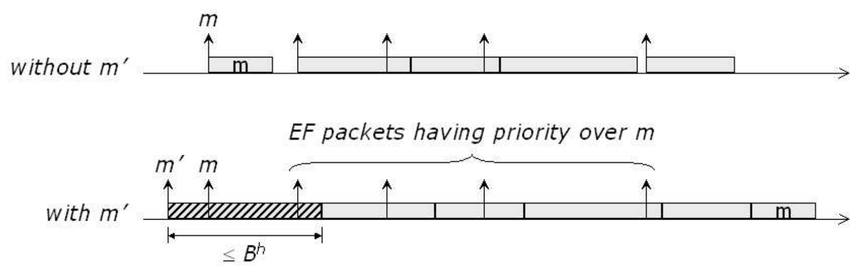

Figure 2: The non-preemptive effect

Property 1 provides an upper bound on the delay incurred by m directly due to m'.

Property 1 *Let τ_i be an* EF *flow following line $\mathcal{L}$ consisting of q nodes numbered from* 1 *to q. The maximum delay directly due to packets not belonging to the* EF *class and incurred by any packet of τ_i is bounded by: $\sum_{h=1}^{q}(B^h - 1)$, where B^h denotes the maximum processing time at node h of any packet of flow not belonging to the* EF *class.*

4.2.2 End-to-end jitter

In a DiffServ domain, packets of any EF flow τ_i experience different end-to-end response times, resulting from varying network delays and varying sojourn times in the visited nodes. The end-to-end jitter of τ_i is the difference between the maximum and minimum end-to-end response times experienced by its packets. Hence, on any line $\mathcal{L}$ followed by τ_i, we have: $Jout_i^{\mathcal{L}} = Rmax_i^{\mathcal{L}} - Rmin_i^{\mathcal{L}}$.

5 The studied techniques

We now compare worst case end-to-end response time and jitter of any EF flow in a DiffServ domain with and without traffic shaping. We recall in Section 5.1 that traffic is continually distorted when no shaping is done and show the consequences of such a distortion on any EF flow in terms of jitter and bursts. In Section 5.2, we detail the jitter cancellation and token bucket techniques and present results on the worst case end-to-end response time and jitter of any EF flow when such techniques are used. These results are established whatever the scheduling policy in the visited nodes. In Section 6, we will see how to apply these results to FIFO scheduling for the EF class.

5.1 Without shaping

As detailed in Section 4.2, packets of any EF flow τ_i can experience variable network delays and sojourn times on each visited node. The difference between the maximum and minimum end-to-end response times experienced by packets of τ_i increases with the number of visited nodes. Then, there is jitter accumulation along the line of a flow. One of the consequences of such a distortion is that bursts of packets may happen along the line and these bursts are more and more important.

On Figure 3, as flows belonging to the EF class enter the domain without jitter, the inter-arrival time between two successive packets of τ_i is equal to T_i. The jitter of τ_i when leaving its source node is: $Jout_i^1 = Rmax_i^1 - Rmin_i^1$. The jitter of τ_i when arriving in the second visited node is equal to this quantity plus the maximum network delay variation, that is: $Jin_i^2 = Rmax_i^1 - Rmin_i^1 + Pmax - Pmin$, and so on.

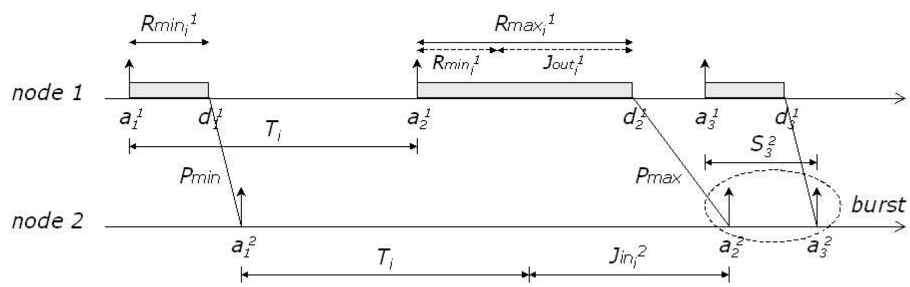

Figure 3: Distortion of any EF flow τ_i on the two first visited nodes

Hence, in any visited node h but the source node, packets of τ_i arrive without necessarily meeting the minimum inter-arrival time of the flow since the inter-arrival time of two successive packets may be equal to: $max\left(0\,; T_i - Jin_i^h\right)$, that is less than T_i. The result of this is that bursts may appear.

Moreover, if we are interested, for example, in a video on demand multimedia system, the distortion can also lead to data shortage. Indeed, as illustrated by Figure 3, the inter-arrival time of two successive packets, periodic on node 1, becomes equal to $T_i + Jin_i^2$ on node 2, possibly leading to data shortage at the client having requested the visualization of a video content. This requires to increase the buffer size of the client.

5.2 With shaping per flow

In order to limit or cancel behaviors described in the previous section, flows belonging to the EF class have to be reshaped. We focus on two shaping techniques: shaping by jitter cancellation and shaping by token bucket.

5.2.1 Shaping by jitter cancellation

As explained in Section 3, the jitter cancellation technique consists in cancelling on each node the jitter of a packet before considering it by the node scheduler. More precisely, any EF packet processed in a node applying the jitter cancellation technique for the EF class will be considered by the node scheduler at its latest possible arrival time (see Figure 4). Each node having its own clock, this technique requires that clocks have a bounded drift and are ε-synchronized. For the sake of simplicity, we do not consider ε in the following analysis. However, the reader can refer to [GM97], where we show how to account for clock precision.

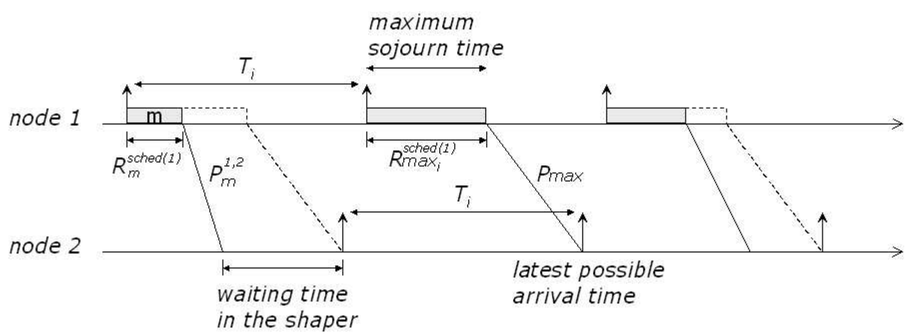

Figure 4: Jitter cancellation principle

Property 2 provides the worst case end-to-end response time and jitter of any EF flow in the DiffServ domain considered when the jitter cancellation technique is used in the EF class.

Property 2 *Let τ_i be a flow belonging to the* EF *class and following line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes. If the jitter cancellation technique is applied to reshape flows belonging to the* EF *class, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot Pmax \\ Jout_i^{\mathcal{L}} = Rmax_i^{sched(q)} - C_i^q. \end{cases}$$

5.2.2 Shaping by token bucket

Figure 5 represents a shaper per flow composed of a token bucket (σ, ρ), where σ denotes the bucket size and ρ the token throughput. A token bucket (σ, ρ) works as follows. The bucket fills up with ρ tokens every second. If σ tokens are already in the bucket, extra-tokens are eliminated. To pass through the shaper, any packet needs a number of tokens corresponding to its length. As long as there is not enough tokens in the bucket to let the packet pass through the shaper, the packet is hold.

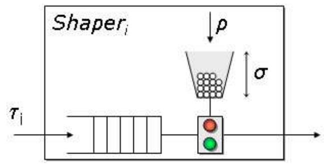

Figure 5: Token bucket principle

Let τ_i be a sporadic flow reshaped by a token bucket (σ, ρ). The following properties provide:

- the necessary configuration of (σ, ρ) for cancelling the bursts of τ_i;
- the maximum delay experienced by τ_i when reshaped by a well-configured shaper;
- the additional jitter introduced by the shaper to the flow;
- the inter-departure time from the shaper of any visited node when the flow generates packets at its maximum rate.

Property 3 *When a sporadic flow τ_i is reshaped on any node h by a token bucket (σ, ρ), where $\sigma = L_i$ and $\rho = L_i/T_i$, its possible bursts are cancelled. Hence, the inter-departure time from the shaper of two successive packets $(m, m+1)$ of τ_i is at least equal to T_i. More precisely, we have:*

$$\sigma = L_i \text{ and } \rho = \frac{L_i}{T_i} \Rightarrow \begin{cases} T_i \leq d_{m+1}^{shap(h)} - d_m^{shap(h)} \leq a_{m+1}^h - a_m^h & \text{if } a_{m+1}^h - a_m^h \geq T_i\,; \\ d_{m+1}^{shap(h)} - d_m^{shap(h)} = T_i & \text{otherwise.} \end{cases}$$

When a token bucket is used to reshape a flow, its parameters have to be those given by Property 3 in order to cancel the bursts of the flow, without increasing the inter-departure time of two successive packets arrived in the shaper with respect to the minimum inter-arrival time of the flow.

Property 4 gives a bound on the maximum delay experienced by any packet of τ_i, when reshaped by a token bucket $(L_i, L_i/T_i)$.

Property 4 *Let τ_i be an* EF *flow following a line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes applying the token bucket technique to reshape* EF *flows. If, on each visited node, the parameters of the shaper associated to τ_i are $(L_i, L_i/T_i)$, then the maximum delay experienced in the node shaper by any packet of τ_i is bounded by the jitter of τ_i when entering the shaper. More precisely, we have:*

$$\forall \text{ packet } m \text{ of } \tau_i,\ \sigma = L_i \text{ and } \rho = \frac{L_i}{T_i} \Rightarrow R_m^{shap(h)} \leq Smax_i^h - S_m^h,$$

this bound being reached when flow τ_i generates packets at its maximum rate.

It is important to notice that reshaping a flow τ_i with a token bucket $(L_i, L_i/T_i)$ does not introduce any additional jitter to the flow, as shown by the following property.

Property 5 *When a sporadic flow τ_i is reshaped on any node h by a token bucket $(L_i, L_i/T_i)$, the node shaper leaves the jitter of τ_i unchanged.*

$$\forall\ packet\ m\ \ of\ \tau_i,\ Smin_i^h \leq S_m^h + R_m^{shap(h)} \leq Smax_i^h.$$

Finally, if τ_i generates packets at its maximum rate, then as soon as a packet of τ_i experiences the maximum delay between its arrival time in the DiffServ domain and its arrival time in the shaper of any visited node h, the inter-departure time from this shaper of two successive packets of τ_i is equal to the minimum inter-arrival time T_i (see Figure 6).

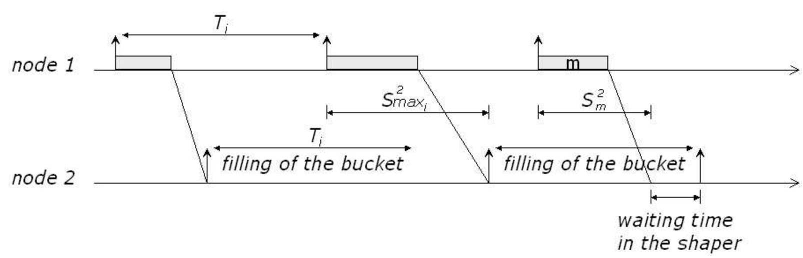

Figure 6: Token bucket effect

Property 6 *Let τ_i be a flow belonging to the* EF *class reshaped by a token bucket $(L_i, L_i/T_i)$ on each visited node. If τ_i generates packets at its maximum rate, then as soon as a packet m' of τ_i experiences the maximum delay between its arrival time in the DiffServ domain and its arrival time in the shaper of any visited node h, the k^{th} packet of τ_i arrived after m' in node h leaves the node shaper exactly $k \cdot T_i$ units of time after m did.*

$$\forall\ packet\ m'\ \ of\ \tau_i,\ S_{m'}^h = Smax_i^h \ \Rightarrow\ \forall\, k \in \mathbb{N},\ \ d_{m'+k}^{shap(h)} - d_{m'}^{shap(h)} = k \cdot T_i.$$

We now evaluate the worst case end-to-end response time and jitter of any EF flow in a DiffServ domain when all nodes apply the token bucket technique to reshape EF flows.

Property 7 *Let τ_i be a flow belonging to the* EF *class and following line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes applying the token bucket technique to reshape flows belonging to the* EF *class. If, on each visited node, the parameters of the token bucket associated to τ_i are $(L_i, L_i/T_i)$, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max} \\ \\ Jout_i^{\mathcal{L}} = \sum_{h=1}^{q} (Rmax_i^{sched(h)} - C_i^h) + (q-1) \cdot (P_{max} - P_{min}). \end{cases}$$

5.3 Example

It is important to notice that for any EF flow τ_i following line $\mathcal{L}$ consisting of q nodes, if τ_i generates packets at its maximum rate, then the worst case end-to-end response time of τ_i is equal to: $Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max}$, whatever the shaping technique.

Indeed, in any node h, as soon as a packet of τ_i experiences the maximum delay between its arrival time in the source node and its arrival time in node h, jitter cancellation technique and token bucket technique are similar. This is illustrated by the following example.

We consider a flow τ_i belonging to the EF class with a minimum inter-arrival time $T_i = 5$. Any packet m of τ_i arrives in node h after S_m^h units of time. We assume that $Jin_i^h = 7$, with $Smin_i^h = 3$ and $Smax_i^h = 10$. As illustrated by Figure 7, when leaving the shaper of node h, any packet arrived after the eighth packet in this node (the first packet such that $S_m^h = Smax_i^h$) experiences from its arrival time in the source node the same response time in both techniques. Then, in term of end-to-end response time, these techniques are equivalent.

On the other hand, in term of end-to-end jitter, the jitter cancellation technique provides better results than the token bucket technique. Indeed, if we consider only packets arrived after the one having experienced the maximum delay between its arrival time in the source node and its arrival time in node h, the jitter of the flow is null when leaving the shaper of node h, whatever the shaping technique (as explained in Section 5.2.2). In our example, all the packets of τ_i arriving after the eighth one in node h experience the same delay between their arrival times in the source node and their departure times from the shaper of node h. But if we consider all the packets of the flow, the end-to-end jitter of τ_i is worse when the token bucket technique is applied, as we can see in Figure 7.

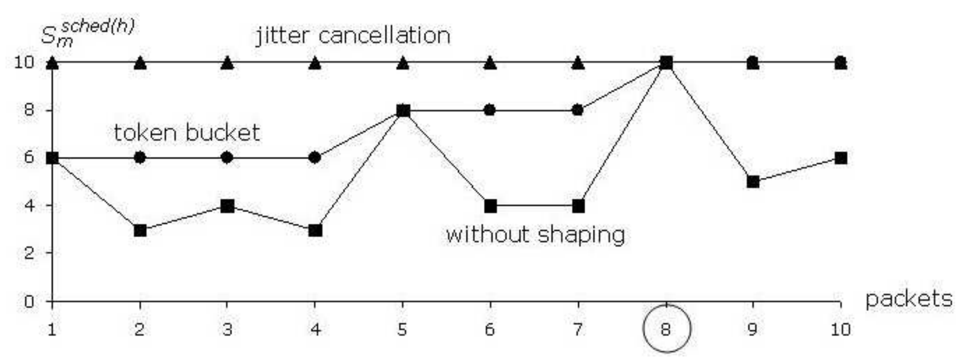

Figure 7: Response times with and without traffic shaping

6 Comparative evaluation when the EF class is scheduled FIFO

In this section, we determine the worst case end-to-end response time and jitter of any EF flow in a DiffServ domain with and without traffic shaping, when the scheduling of the EF class is FIFO. Before focusing on each of the three techniques considered in this paper, we give the following property, concerning the non-preemptive effect on the EF class when packets in this class are served FIFO.

Property 8 *Let τ_i be a flow belonging to the EF class. Let $\mathcal{L}$ be a line followed by τ_i, it consists of q nodes numbered from 1 to q. If the scheduling of the EF class is FIFO, then the delay due to the non-preemption is bounded by: $\sum_{h=1}^{q}(B^h - 1)$, where B^h denotes the maximum processing time at node h of any packet of flow not belonging to the EF class.*

6.1 Without shaping

We focus on the worst case end-to-end response time and jitter of any EF flow in a DiffServ domain, when nodes schedule the EF flows according to FIFO. Sections 6.1.1 and 6.1.2 recall known results for this scheduling policy when all the EF flows (i) are processed in a single node and (ii) follow the same line in the DiffServ domain. Finally, we extend these results to the general case, that is all the EF flows follow different lines.

6.1.1 The single node case

The two following properties give the worst case response time of any EF flow when all the flows belonging to the EF class are processed in a single node, in the presence of other classes flows. EF packets are assumed to be released without jitter in the former and with jitter in the latter.

Property 9 *In the uniprocessor context, if the scheduling of the EF class is FIFO and the EF flows release packets without jitter, then the worst case response time of any EF flow τ_i, in the presence of other classes flows, is obtained in the first busy period of the synchronous scenario, that is packets of the EF flows are released at the same time, and is equal to: $R_{max_i} = \sum_{j=1}^{n} C_j + (B - 1)$.*

Property 10 *In the uniprocessor context, if the scheduling of the EF class is FIFO and the EF flows release packets with jitter, the worst case response time of any EF flow τ_i, in the presence of other classes flows, is equal to:*

$$R_{max_i} = \max_{-Jin_i \leq t < L} \left\{ \sum^{n}{}_{\substack{j=1 \\ t \geq -Jin_j}} \left(1 + \left\lfloor \frac{t + Jin_j}{T_j} \right\rfloor \right) \cdot C_j - t \right\} + (B - 1),$$

where J_{in_j} is the release jitter of EF flow τ_j and L is the length of the longest busy period, first solution of $L = \sum_{j=1}^{n} \lceil (L + J_{in_j})/T_j \rceil \cdot C_j$.

6.1.2 Extension to the single line case

An extension of the results known in the uniprocessor context has been studied in [GMM02], providing a bound on the worst case end-to-end response time of any EF flow in a DiffServ domain, when all the EF flows follow the same line. This bound is given in the following property.

Property 11 *Let $\tau = \{\tau_1, \tau_2, ..., \tau_n\}$ be the set of the n sporadic* EF *flows following the same line $\mathcal{L}$. If the scheduling of the* EF *class is* FIFO*, then we have:*

$$\forall \tau_i \in \tau,\ R_i^{\mathcal{L}} \leq \sum_{j=1}^{n} C_j^{slow} + \sum_{\substack{h=1 \\ h \neq slow}}^{q} C_{max}^h + \sum_{h=1}^{q}(B^h - 1) + (q-1) \cdot P_{max},$$

where slow denotes the slowest node among the q visited nodes and $C_{max}^h = \max_{j=1..n}\{C_j^h\}$.

It is important to notice that this bound can be reached when the q nodes are visited in decreasing order of their processing capacities. Then, the worst case end-to-end response time of any EF flow τ_i meets: $R_i^{\mathcal{L}} = \sum_{j=1}^{n} C_j^q + \sum_{h=1}^{q-1} C_{max}^h + \sum_{h=1}^{q}(B^h - 1) + (q-1) \cdot P_{max}$.

6.1.3 The general case

The following property extends the results given in Section 6.1.1 to the general case, that is EF flows follow different lines in the DiffServ domain. For the sake of simplicity, we first consider that any EF flow τ_j following line $\mathcal{L}'$ with $\mathcal{L}' \neq \mathcal{L}$ and $\mathcal{L}' \bigcap \mathcal{L} \neq \emptyset$ never visits a node of line $\mathcal{L}$ after having left line $\mathcal{L}$ (cf. Assumption 1).

Assumption 1 *For any* EF *flow τ_j following line $\mathcal{L}'$ with $\mathcal{L}' \neq \mathcal{L}$ and $\mathcal{L}' \bigcap \mathcal{L} \neq \emptyset$, if there exists a node $h \in \mathcal{L}' \bigcap \mathcal{L}$ such that τ_j visits $h' \neq h+1$ immediately after h, then τ_j never visits a node $h'' \in \mathcal{L}$ after.*

To remove this assumption, we decompose an EF flow in as many independent flows as needed to meet Assumption 1. To achieve that, the idea is to consider an EF flow crossing line $\mathcal{L}$ after it left $\mathcal{L}$ as a new EF flow. We proceed by iteration until meeting Assumption 1. We then apply Property 12 considering all these flows.

Property 12 *If the condition $\sum_{h=1}^{q} \sum_{\substack{j=1 \\ first_j = h}}^{n} (C_j^{slow_j}/T_j) \leq 1$ is met, then the worst case end-to-end response time of any* EF *flow τ_i following line $\mathcal{L}$ meets:*

$$Rmax_i^{\mathcal{L}} \leq \sum_{h=1}^{q} \left(\sum_{\substack{j=1 \\ first_j = h}}^{n} \left(1 + \frac{Smax_i^h + Jin_j^h}{T_j}\right) \cdot C_j^{slow_j} \right) + \sum_{\substack{h=1 \\ h \neq slow_i}}^{q} C_{max}^h + \sum_{h=1}^{q}(B^h - 1) + (q-1) \cdot P_{max},$$

where $slow_j$ is the slowest node visited by τ_j on line $\mathcal{L}$ and $first_j$ is the first node visited by τ_j on line $\mathcal{L}$.

In [MMG03], we show how to bound S_m^h, the maximum delay experienced by packets of flow τ_i from their arrival times in the domain to reach node h and Jin_j^h, the delay jitter experienced by flow τ_j between its source node and node h.

The first term of the bound given in Property 12 depends on the flows following or crossing line $\mathcal{L}$. More precisely, it is proportional to their processing time on their slowest node on $\mathcal{L}$. The other terms of the given bound depend on the number of nodes visited by τ_i. Moreover, we can notice that this bound is more accurate than the sum of the maximum sojourn times on the visited nodes, plus the sum of the maximum network delays. This shows the interest of the trajectory approach, comparatively to the holistic one.

6.2 With shaping by jitter cancellation

We first focus on the worst case end-to-end scenario of any EF flow when the scheduling of the EF class is FIFO.

Property 13 *Let $\mathcal{D}$ be a DiffServ domain where all nodes reshape flows belonging to the* EF *class applying the jitter cancellation technique. If the scheduling of the* EF *class is* FIFO*, then the worst case end-to-end response time of any* EF *flow is obtained considering the synchronous scenario in each visited node.*

The following property provides the worst case end-to-end response time and jitter of any flow belonging to the EF class when the jitter cancellation technique is used to reshape flows belonging to the EF class and packets in this class are served FIFO.

Property 14 *Let τ_i be a flow belonging to the* EF *class and following line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes applying the jitter cancellation technique to reshape flows belonging to the* EF *class. If the* EF *class is scheduled* FIFO*, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} \sum_{j=1}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot P_{max} \\ Jout_i^{\mathcal{L}} \; = \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^q + (B^q - 1). \end{cases}$$

6.3 With shaping by token bucket

Before computing the worst case end-to-end response time and jitter of any EF flow τ_i in the DiffServ domain when the token bucket technique is used to reshape flows belonging to the EF class, we show how to obtain the worst case end-to-end response time of τ_i.

Property 15 *Let τ_i be a flow belonging to the* EF *class reshaped by a token bucket $(L_i, L_i/T_i)$ on each visited node. If the scheduling of the* EF *class is* FIFO*, then a bound on the worst case end-to-end response time of τ_i is obtained considering the synchronous scenario in each visited node.*

Property 15 allows us to determine a bound on the worst case end-to-end response time and jitter of any EF flow when the scheduling of the EF class is FIFO.

Property 16 *Let τ_i be a flow belonging to the* EF *class reshaped by a token bucket $(L_i, L_i/T_i)$ on each visited node. Let $\mathcal{L}$ be a line followed by τ_i, where $\mathcal{L}$ consists of q nodes. If the scheduling of the* EF *class is* FIFO*, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} \leq \sum_{h=1}^{q} \sum_{j=1}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot P_{max} \\ Jout_i^{\mathcal{L}} \leq \sum_{h=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot (P_{max} - P_{min}), \end{cases}$$

7 Discussion

7.1 Jitter cancellation versus token bucket

As pointed in Section 5.3, the worst case end-to-end response time of any EF flow τ_i that follows a line $\mathcal{L}$ consisting of q nodes is: $Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max}$, whatever the shaping technique. Moreover, if the scheduling of the EF class is FIFO, then the guarantee on the end-to-end response time of τ_i in any visited node scheduler is the same in both techniques.

Therefore, we get: $Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} \sum_{j=1}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot P_{max}$.

As both techniques provide the same deterministic guarantee in term of end-to-end response time, none of them is more appropriate than the other when the QoS requirements concern only this parameter. Nevertheless, the token bucket technique does not require nodes to have ε-synchronized clocks, unlike the jitter cancellation technique (see Section 5.2.1).

On the other hand, in comparing the results given in Properties 14 and 16, we can conclude that the jitter cancellation technique is always more appropriate than the token bucket one when focusing on the end-to-end jitter parameter.

Indeed, for any EF flow τ_i following a line $\mathcal{L}$ that consists of q nodes, the guarantee on the worst case end-to-end jitter experienced by τ_i when applying the token bucket technique is equal to: $\sum_{h=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot (P_{max} - P_{min})$, while the worst case end-to-end jitter of τ_i when applying the jitter cancellation technique is equal to: $\sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^q + (B^q - 1)$. Hence, the difference, that increases with the number of visited nodes, is equal to: $\sum_{h=1}^{q-1} \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^h + \sum_{h=1}^{q-1} \left(B^h - 1\right) + (q-1) \cdot (P_{max} - P_{min})$.

7.2 With and without traffic shaping comparison

7.2.1 The single node case

Let us assume that τ_i visits a single node h in the DiffServ domain. If this node does not apply any shaping technique, then the worst case response time of τ_i on node h is greater than: $\sum_{j=1}^{n}\left(1+\lfloor Jin_j^h/T_j\rfloor\right)\cdot C_j^h+(B^h-1)$ (see Property 10). This delay increases when the jitters of the other EF flows visiting this node increase. The jitter cancellation technique, by cancelling the input jitter of all the EF flows in the node considered, would lead to reduce the worst case response time of τ_i on node h to: $\sum_{j=1}^{n} C_j^h$. The token bucket technique, by cancelling the bursts of all the EF flows in the node considered, would lead to the same result when EF flows generate packets at their maximum rates.

7.2.2 The single line case

As in the preceding section, the QoS requirements allow to select the most appropriate technique. If the end-to-end response time is the parameter at the utmost importance, the no shaping technique is better than any shaping technique. Indeed, when the EF class is scheduled FIFO, the worst case end-to-end response time experienced by any EF flow τ_i when applying any of the shaping techniques is equal to: $\sum_{h=1}^{q}\sum_{j=1}^{n} C_j^h+\sum_{h=1}^{q}\left(B^h-1\right)+(q-1)\cdot P_{max}$. Without shaping, the worst case end-to-end response time of the EF flow τ_i is bounded by: $\sum_{j=1}^{n} C_j^{slow}+\sum_{\substack{h=1\\ h\neq slow}}^{q} C_{max}^h+\sum_{h=1}^{q}(B^h-1)+(q-1)\cdot P_{max}$, where *slow* denotes the slowest node among the q visited nodes of line $\mathcal{L}$ and $C_{max}^h=\max_{j=1..n}\{C_j^h\}$. The difference is then equal to: $\sum_{\substack{h=1\\ h\neq slow}}^{q}(\sum_{j=1}^{n}(C_j^h)-C_{max}^h)$, that increases with the number of visited nodes. Hence, the studied shaping techniques will never be appropriate when all the EF flows follow the same line, whatever the number of visited nodes or the number of EF flows.

Although the no shaping technique is the most appropriate in the single line case in term of end-to-end response time, this is to the detriment of the second QoS parameter: the end-to-end jitter. Indeed, the jitter cancellation technique provides the best results in term of deterministic guarantee on the end-to-end jitter. But it is interesting to underline that the no shaping technique leads to a jitter, for any EF flow τ_i, smaller than the one obtained when EF flows are reshaped by token buckets, since τ_i experiences the same minimum end-to-end response time but different maximum end-to-end response times along the line $\mathcal{L}$.

Nevertheless, as pointed in Section 5.1, it can be useful, and even essential to cancel bursts of a flow on the visited nodes. Moreover, as the no shaping technique leads to a smaller worst case end-to-end response time than shaping with token bucket, the minimum end-to-end response time being the same in both techniques, the worst case end-to-end jitter is smaller with the no shaping technique than with shaping by token bucket.

7.2.3 The general case

In the general case, the EF flows follow different lines. Property 12 extends the results known in the uniprocessor context for the FIFO scheduling policy. Let τ_i be an EF flow following line $\mathcal{L}$ and τ_j, $j \neq i$, an EF flow crossing line $\mathcal{L}$. Let $first_i$ (resp. $first_j$) denote the first node visited by τ_i (resp. τ_j) on line $\mathcal{L}$. Let $slow_i$ (resp. $slow_j$) denote the slowest node visited by τ_i (resp. τ_j) on line $\mathcal{L}$. The difference between the bounds on the end-to-end response time of any EF flow τ_i with and without traffic shaping is equal to:

$$\sum_{h=1}^{q} \left(\sum_{\substack{j=1 \\ h \neq slow_j}}^{n} C_j^h \right) - \sum_{j=1}^{n} \left(\frac{Smax_i^{first_j} + Jin_j^{first_j}}{T_j} \right) \cdot C_j^{slow_j} - \sum_{\substack{h=1 \\ h \neq slow_i}}^{q} C_{max}^h.$$

It is important to notice that the bounds given for the two shaping techniques considered are reached. Hence, the bound on the response time of τ_i is smaller without traffic shaping when:

$$\sum_{j=1}^{n} \left(\frac{Smax_i^{first_j} + Jin_j^{first_j}}{T_j} \right) \cdot C_j^{slow_j} \leq \sum_{h=1}^{q} \left(\sum_{\substack{j=1 \\ h \neq slow_j}}^{n} C_j^h \right) - \sum_{\substack{h=1 \\ h \neq slow_i}}^{q} C_{max}^h.$$

Then, the studied shaping techniques are appropriate not only when τ_i crosses other EF flows on its last nodes, but also when the jitters of the EF flows crossing τ_i are large.

8 Example

In this section, we give an example of bounds on the end-to-end response times of EF flows in a DiffServ domain, with and without traffic shaping. We assume that $\tau = \{\tau_1, \tau_2, \tau_3, \tau_4\}$. As shown in Figure 8, τ_1 visits nodes 1, 2, 3 and 4, τ_2 visits nodes 5, 2, 3 and 6, and τ_3 visits nodes 7, 2 and 8. All the EF flows have a period equal to 10 and enter the domain without jitter. For any packet of any flow, the processing time on any node is equal to 2. Finally, we have $P_{max} = P_{min} = 1$.

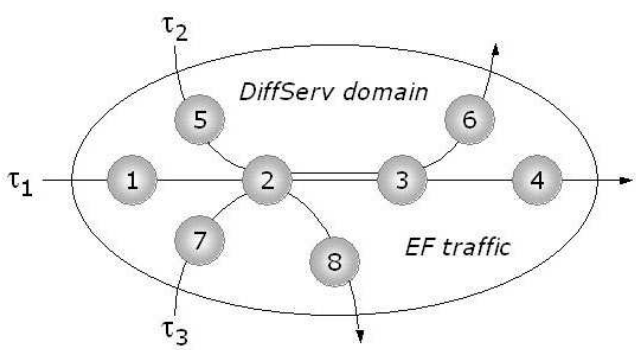

Figure 8: EF traffic in the DiffServ domain

As part of this paper, a simulation tool has been developed for providing the exhaustive solution of a real-time scheduling problem in a network. Indeed, once the different parameters are specified, all the possible scenarios are generated and traffic feasibility is checked for each of them. The simulation result is a file containing the real worst case end-to-end response time of each flow.

The following table gives the worst case end-to-end response time of flow τ_1 for each technique considered (no traffic shaping, jitter cancellation and token bucket). Notice that the condition of Property 12 is met, as $\sum_{h=1}^{q} \sum_{\substack{j=1 \\ first_j=h}}^{n} (C_j^{slow_j}/T_j)) = 0.8 \leq 1$.

Table 1: Worst case response times of flow τ_1

	no traffic shaping	**jitter cancellation**	**token bucket**
by simulation	21	29	29
by computation	23	29	29

Remark: If τ_2 or τ_3 is disturbed by new EF flows not crossing τ_1, the real worst case end-to-end response time of τ_1, just as the bound on this worst case end-to-end response time, will possibly increase when no traffic shaping is used. On the other hand, the worst case end-to-end response time of τ_i will be unchanged if shaping techniques are used.

We can notice that the utilization factor for the EF class reaches 80% on node 2.

9 Conclusion

Varying network delays and sojourn times in the network lead to traffic distortions: the minimum inter-arrival time between two successive packets of the same flow is no more met. To avoid this, traffic shaping has been introduced. We have focused on three techniques: no shaping, shaping with jitter cancellation and shaping with token bucket. We have first established results independent of the scheduling policy used in the node. We have then computed a bound on the end-to-end response time and a bound on the end-to-end jitter that can be deterministically guaranteed to EF flows when scheduled FIFO, with and without traffic shaping. In the case of a single line for all flows, the no shaping technique performs better in term of end-to-end response time. The end-to-end response times obtained by jitter cancellation and token bucket are equivalent. Jitter cancellation is always more appropriate to meet a constraint on the end-to-end jitter. The jitter obtained with the no shaping technique is better than with token bucket. In the general case, we have established a condition under which the bound on the end-to-end response time of any EF flow is better without traffic shaping. Moreover, jitter cancellation and token bucket provide the same worst case end-to-end response time. Nevertheless, jitter cancellation always provides a smaller end-to-end jitter.

References

[BBC01] J. Bennett, K. Benson, A. Charny and al., « Delay jitter bounds and packet scale rate guarantee for Expedited Forwarding », INFOCOM'2001, Anchorage, USA, April 2001.

[BBC98] S. Blake, D. Black, M. Carlson and al., « An architecture for Differentiated Services », RFC 2475, December 1998.

[CLB00] A. Charny, J. le Boudec, « Delay bounds in a network with aggregate scheduling », QoFIS, Berlin, Germany, October 2000.

[CS98] F. Chiussi, V. Sivaraman, « Achieving high utilization in guaranteed services networks using early-deadline-first scheduling », IWQoS'98, Napo, USA, May 1998.

[GEO98] L. George, « Ordonnancement en-ligne temps réel critique dans les systèmes distribués », PhD. thesis, University of Versailles-St Quentin, France, January 1998.

[GKM01] L. George, S. Kamoun, P. Minet, « First come first served: some results for real-time scheduling », PDCS'01, Dallas, USA, August 2001.

[GMM01] L. George, D. Marinca, P. Minet, « A solution for a deterministic QoS in multimedia systems », International Journal on Computer and Information Science, Vol. 1, Sept. 2001.

[GMM02] L. George, S. Martin, P. Minet, « Ordonnancement FIFO et garantie déterministe de qualité de service en environnement distribué », RTS'2002, Paris, France, March 2002.

[GM97] L. George, P. Minet, « A FIFO worst case analysis for a hard real-time distributed problem with consistency constraints », ICDCS'1997, Baltimore, USA, May 1997.

[GGP96] L. Georgiadis, R. Guérin, V. Peris and al., « Efficient network QoS provisioning based on per node traffic shaping », IEEE ACM Transactions on Networking, Vol. 4, August 1996.

[GCL01] M. Gerla, C. Casetti, S. Lee and al., « Resource allocation and admission control styles on QoS DiffServ networks », QoS-IP 2001, Rome, Italy, 2001.

[HBW99] J. Heinanen, F. Baker, W. Weiss and al., « Assured Forwarding PHB group », RFC 2597, June 1999.

[JNP99] V. Jacobson, K. Nichols, K. Poduri, « An Expedited Forwarding PHB », RFC 2598, June 1999.

[LBT97] J. le Boudec, P. Thiran, « A note on time and space methods in network calculus », Technical Report, N. 97/224, Ecole Polytechnique Fédérale de Lausanne, Swiss, April 1997.

[LFW00] F. le Faucheur, L. Wu, S. Davari and al., « MPLS support of Differentiated Services », Internet draft, August 2000.

[MMG03] S. Martin, P. Minet, L. George, « A DiffServ-MPLS solution offering real-time end-to-end guarantees », INRIA Research Report, N. 4831, May 2003.

[PG94] A. Parekh, R. Gallager, « A generalized processor sharing approach to flow control in integrated services networks: the multiple node case », IEEE ACM Transactions on Networking, Vol. 2, April 1994.

[RVC01] E. Rosen, A. Viswanathan, R. Callon, « Multiprotocol Label Switching Architecture », RFC 3031, January 2001.

[SC00] V. Sivaraman, F. Chiussi, « Providing end-to-end statistical delay guarantees with Earliest Deadline First scheduling and per-hop traffic shaping », INFOCOM'00, Proc. IEEE, Vol. 2 , pp. 631-640.

[SCG00] V. Sivaraman, F. Chiussi, M. Gerla, « Traffic shaping for end-to-end delay guarantees with EDF scheduling », IWQoS'2000, Pittsburgh, June 2000.

[SCG01] V. Sivaraman, F. Chiussi, M. Gerla, « End-to-end statistical delay service under GPS and EDF scheduling: a comparaison study », INFOCOM'2001, Anchorage, Alaska, April 2001.

[TC94] K. Tindell, J. Clark, « Holistic schedulability analysis for distributed hard real-time systems », Microprocessors and Microprogramming, Euromicro Journal, Vol. 40, pp. 117-134, 1994.

[VLB02] M. Vojnovic, J. le Boudec, « Stochastic analysis of some expedited forwarding networks », INFOCOM'2002, New York, USA, June 2002.

Annex

Property 1 *Let τ_i be a flow belonging to the* EF *class. Let $\mathcal{L}$ be a line followed by τ_i, it consists of q nodes numbered from 1 to q. The maximum delay directly due to packets not belonging to the* EF *class and incurred by any packet of τ_i is bounded by: $\sum_{h=1}^{q}(B^h - 1)$, where B^h denotes the maximum processing time at node h of any packet of flow not belonging to the* EF *class.*

Proof: Let us show that if m is delayed on any node h of line $\mathcal{L}$ by a packet m' not belonging to the EF class, then in the worst case the packet m arrives on the scheduler of h just after the execution of m' begins. Indeed, by contradiction, if m arrives on the node scheduler before or at the same time as m', then as m belongs to the EF class and this class has the highest priority, m will be processed before m'. Hence a contradiction. Thus, in the worst case, m arrives on the scheduler of h one time unit after the execution of m' begins, because we assume a discrete time.

The maximum processing time for m' on h is B^h. By convention, if all packets visiting the node h belong to the EF class, we note $B^h - 1 = 0$. Hence, in any case, the maximum delay incurred by m on h directly due to a packet not belonging to the EF class is equal to $B^h - 1$. As we make no particular assumption concerning the classes other than the EF class, in the worst case packet m is delayed for $B^h - 1$ on any visited node h. ■

Property 2 *Let τ_i be a flow belonging to the* EF *class and following line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes. If the jitter cancellation technique is applied to reshape flows belonging to the* EF *class, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot Pmax \\ Jout_i^{\mathcal{L}} = Rmax_i^{sched(q)} - C_i^q. \end{cases}$$

Proof: Let τ_i be a flow belonging to the EF class and following line $\mathcal{L}$. We assume that $\mathcal{L}$ consists of q nodes applying the jitter cancellation technique for the EF class. As there is no jitter on any source node, such a node schedules EF packets as soon as they arrive. In any other node $h \in \mathcal{L}$, packets of τ_i are hold in the node shaper until their latest possible reception times. Thus, their arrival times in the node scheduler are the same as those in the source node.

Therefore, τ_i arrives in node h with a jitter depending only on the jitter introduced by the scheduler of the previous node and the link between the two nodes (see Figure 4). The delay incurred by any packet m of τ_i in the shaper of node $h \in [2, q]$ is then equal to:

$$R_m^{shap(h)} = \left(Rmax_i^{sched(h-1)} + Pmax\right) - \left(R_m^{sched(h-1)} + P_m^{h-1,h}\right).$$

Hence, the end-to-end response time of packet m is equal to:

$$\begin{aligned}
R_m^{\mathcal{L}} &= \sum_{h=1}^{q} R_m^{shap(h)} + \sum_{h=1}^{q} R_m^{sched(h)} + \sum_{h=1}^{q-1} P_m^{h,h+1} \\
&= \sum_{h=2}^{q} \left(Rmax_i^{sched(h-1)} + Pmax - R_m^{sched(h-1)} - P_m^{h-1,h} \right) + \sum_{h=1}^{q} R_m^{sched(h)} + \sum_{h=1}^{q-1} P_m^{h,h+1} \\
&= \sum_{h=1}^{q-1} Rmax_i^{sched(h)} + (q-1) \cdot Pmax + R_m^{sched(q)}.
\end{aligned}$$

Then, we get: $Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot Pmax$.

Concerning the end-to-end jitter of flow τ_i, as the jitter cancellation technique consists in cancelling on each node the jitter of a flow before it is considered by the node scheduler, the delay jitter of τ_i is null when the flow arrives in the scheduler of node q. Then, the end-to-end jitter of τ_i depends on the scheduler of the last visited node.

Since the response time of any packet of τ_i in this scheduler is at most equal to $Rmax_i^{sched(q)}$ and at least equal to C_i^q, we get: $Jout_i^{\mathcal{L}} = Rmax_i^{sched(q)} - C_i^q$. ■

Lemma 1 *Let τ_i be a sporadic flow reshaped on any node h by a token bucket $(L_i, L_i/T_i)$, where L_i is the length of any packet of τ_i and T_i is the minimum inter-arrival time between two successive packets of τ_i. The response time of packet m in the shaper of node h is equal to: $R_m^{shap(h)} = x_m^h \cdot \frac{T_i}{L_i}$, where x_m^h denotes the number of tokens to generate at time a_m^h in order to let the m^{th} packet of τ_i pass through the shaper of node h and is equal to:*

$$x_m^h = \max\left(0; x_{m-1}^h + L_i - (a_m^h - a_{m-1}^h) \cdot \tfrac{L_i}{T_i}\right) = \max_{k=0..m-1} \left(k \cdot L_i - (a_m^h - a_{m-k}^h) \cdot \tfrac{L_i}{T_i}\right).$$

Proof: By definition, x_m^h is equal to the number of tokens that had to be generated at time a_{m-1}^h in order to let the $(m-1)^{th}$ packet of τ_i pass through the node shaper, plus L_i tokens consumed by packet $(m-1)$, minus the number of tokens generated by the token bucket during the interval $[a_{m-1}^h, a_m^h]$. If x_m^h is less than 0, that means there are more tokens than necessary. In this case, $|x_m^h|$ represents the number of extra-tokens. Moreover, when the m^{th} packet of τ_i enters the shaper of node h, there are at most $|L_i - \sigma|$ tokens unnecessary if L_i is less than or equal to σ and at least $L_i - \sigma$ tokens to generate otherwise. Hence, we get: $x_m^h = \max\left(L_i - \sigma\,; x_{m-1}^h + L_i - (a_m^h - a_{m-1}^h) \cdot \rho\right)$. If we recursively substitute the x_j^h terms, where $j = m-1..1$, for their respective values, we finally get: $x_m^h = \max_{k=0..m-1} \left(L_i - \sigma + k \cdot L_i - (a_m^h - a_{m-k}^h) \cdot \rho\right)$.

On the other hand, if $x_m^h \geq 0$, then packet m has to wait (x_m^h/ρ) units of time to pass through the node shaper. Hence, the response time of packet m in the shaper of node h is equal to: $R_m^{shap(h)} = \max\left(0\,; \frac{x_m^h}{\rho}\right)$.

Therefore, if the token bucket parameters are (L_i, T_i), we get Lemma 1. ■

Property 3 *When a sporadic flow τ_i is reshaped on any node h by a token bucket (σ, ρ), where $\sigma = L_i$ and $\rho = L_i/T_i$, its possible bursts are cancelled. Hence, the inter-departure time from the shaper of two successive packets $(m, m+1)$ of τ_i is at least equal to T_i. More precisely, we have:*

$$\sigma = L_i \text{ and } \rho = \tfrac{L_i}{T_i} \Rightarrow \begin{cases} T_i \leq d_{m+1}^{shap(h)} - d_m^{shap(h)} \leq a_{m+1}^h - a_m^h & \text{if } a_{m+1}^h - a_m^h \geq T_i\,; \\ d_{m+1}^{shap(h)} - d_m^{shap(h)} = T_i & \text{otherwise.} \end{cases}$$

Proof: Let τ_i be a flow reshaped on any node h by a token bucket $(L_i, L_i/T_i)$. The delay between the departure times from the shaper of any node h of two successive packets $(m, m+1)$ of τ_i is equal to: $d_{m+1}^{shap(h)} - d_m^{shap(h)} = a_{m+1}^h + R_{m+1}^{shap(h)} - (a_m^h + R_m^{shap(h)})$. By Lemma 1, we get: $d_{m+1}^{shap(h)} - d_m^{shap(h)} = a_{m+1}^h - a_m^h + x_{m+1}^h \cdot \frac{T_i}{L_i} - x_m^h \cdot \frac{T_i}{L_i}$. Hence, we have:

$$\begin{aligned} d_{m+1}^{shap(h)} - d_m^{shap(h)} &= a_{m+1}^h - a_m^h + \max\left(0\,; x_m^h + L_i - (a_{m+1}^h - a_m^h) \cdot \tfrac{L_i}{T_i}\right) \cdot \tfrac{T_i}{L_i} - x_m^h \cdot \tfrac{T_i}{L_i} \\ &\geq a_{m+1}^h - a_m^h + x_m^h \cdot \tfrac{T_i}{L_i} + T_i - (a_{m+1}^h - a_m^h) - x_m^h \cdot \tfrac{T_i}{L_i} \;=\; T_i. \end{aligned}$$

To compute an upper bound on $d_{m+1}^{shap(h)} - d_m^{shap(h)}$, we have to distinguish two cases:

- if $(a_{m+1}^h - a_m^h)$ is less than or equal to T_i, then: $x_m^h + L_i - (a_{m+1}^h - a_m^h) \cdot \frac{L_i}{T_i} \geq 0$. Thus, we have: $d_{m+1}^{shap(h)} - d_m^{shap(h)} = a_{m+1}^h - a_m^h + x_m^h \cdot \frac{T_i}{L_i} + T_i - (a_{m+1}^h - a_m^h) - x_m^h \cdot \frac{T_i}{L_i} = T_i$.
- if $(a_{m+1}^h - a_m^h)$ is greater than or equal to T_i, then $L_i - (a_{m+1}^h - a_m^h) \cdot \frac{L_i}{T_i} \leq 0$. Hence:
  - if $x_m^h + L_i - (a_{m+1}^h - a_m^h) \cdot \frac{L_i}{T_i} \geq 0 \;\Rightarrow\; d_{m+1}^{shap(h)} - d_m^{shap(h)} = T_i \leq a_{m+1}^h - a_m^h$;
  - if $x_m^h + L_i - (a_{m+1}^h - a_m^h) \cdot \frac{L_i}{T_i} \leq 0 \;\Rightarrow\; d_{m+1}^{shap(h)} - d_m^{shap(h)} \leq a_{m+1}^h - a_m^h$. ■

Property 4 *Let τ_i be a flow belonging to the* EF *class and following a line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes applying the token bucket technique to reshape flows belonging to the* EF *class. If, on each visited node, the parameters of the shaper associated to τ_i are $(L_i, L_i/T_i)$, then the maximum delay experienced in the node shaper by any packet of τ_i is bounded by the jitter of τ_i when entering the shaper. More precisely, we have:*

$$\forall \text{ packet } m \text{ of } \tau_i,\ \sigma = L_i \text{ and } \rho = \tfrac{L_i}{T_i} \;\Rightarrow\; R_m^{shap(h)} \leq Smax_i^h - S_m^h,$$

this bound being reached when flow τ_i generates packets at its maximum rate.

Proof: Let τ_i be a flow reshaped on any node h by a token bucket $(L_i, L_i/T_i)$. By Lemma 1, the number of tokens to generate at time a_m^h in order to let the m^{th} packet of τ_i pass through the shaper of node h is equal to: $x_m^h = \max_{k=0..m-1}(k \cdot L_i - (a_m^h - a_{m-k}^h) \cdot \frac{L_i}{T_i})$. Besides, we have: $a_m^h - a_{m-k}^h = (a_m^1 + S_m^h) - (a_{m-k}^1 + S_{m-k}^h) \geq k \cdot T_i + S_m^h - S_{m-k}^h$. Hence, we get $x_m^h \geq \max_{k=0..m-1}((k \cdot T_i + S_m^h - S_{m-k}^h) \cdot \frac{L_i}{T_i}) \leq \frac{L_i}{T_i} \cdot \max_{k=0..m-1}\left(S_{m-k}^h - S_m^h\right)$.

Moreover, as soon as a packet m' of τ_i experiences the maximum delay between its arrival time in the source node and its arrival time in node h, we have for any packet $m \geq m'$: $\frac{L_i}{T_i} \cdot \max_{k=0..m-1}\left(S_{m-k}^h - S_m^h\right) = \frac{L_i}{T_i} \cdot (Smax_i^h - S_m^h)$. Therefore, the delay incurred by any packet m of τ_i in the shaper of node h is equal to: $R_m^{shap(h)} = x_m^h \cdot \frac{T_i}{L_i} \leq Smax_i^h - S_m^h$. This bound is reached for all the packets arrived in the shaper after the one having experienced the maximum delay to reach node h, if flow τ_i generates packets at its maximum rate. ■

Property 5 *When a sporadic flow τ_i is reshaped on any node h by a token bucket $(L_i, L_i/T_i)$, the node shaper leaves the jitter of τ_i unchanged.*

$$\forall \text{ packet } m \text{ of } \tau_i,\ Smin_i^h \leq S_m^h + R_m^{shap(h)} \leq Smax_i^h.$$

Proof: Let τ_i be a flow belonging to the EF class. For any packet m of τ_i, the delay between its arrival time in the DiffServ domain and its departure time from the shaper of any node h is equal to: $d_m^{shap(h)} - a_m^1 = S_m^h + R_m^{shap(h)}$. If $\sigma = L_i$ and $\rho = L_i/T_i$, then $R_m^{shap(h)} \leq Smax_i^h - S_m^h$ (see Property 4). Hence, we have: $S_m^h + R_m^{shap(h)} \leq S_m^h + Smax_i^h - S_m^h$. As $Smin_i^h \leq S_m^h$, we finally get: $Smin_i^h \leq S_m^h + R_m^{shap(h)} \leq Smax_i^h$. ■

Property 6 *Let τ_i be a flow belonging to the* EF *class reshaped by a token bucket $(L_i, L_i/T_i)$ on each visited node. If τ_i generates packets at its maximum rate, then as soon as a packet m' of τ_i experiences the maximum delay between its arrival time in the DiffServ domain and its arrival time in the shaper of any visited node h, the k^{th} packet of τ_i arrived after m' in node h leaves the node shaper exactly $k \cdot T_i$ units of time after m did.*

$$\forall \text{ packet } m' \text{ of } \tau_i,\ S_{m'}^h = Smax_i^h \Rightarrow \forall k \in \mathbb{N},\ d_{m'+k}^{shap(h)} - d_{m'}^{shap(h)} = k \cdot T_i.$$

Proof: Let τ_i be a flow belonging to the EF class and following line $\mathcal{L}$. For all packets m and $m+1$ of τ_i, we have: $\forall\ h \in \mathcal{L},\ (d_{m+1}^{shap(h)} - a_{m+1}^1) - (d_m^{shap(h)} - a_m^1) = d_{m+1}^{shap(h)} - d_m^{shap(h)} - T_i$. Moreover, if $\sigma = L_i$ and $\rho = L_i/T_i$, then $d_{m+1}^{shap(h)} - d_m^{shap(h)} \geq T_i$ (see Property 3). Hence, $(d_{m+1}^{shap(h)} - a_{m+1}^1) - (d_m^{shap(h)} - a_m^1) \geq 0$. The delay between the arrival time of any packet of τ_i in the DiffServ domain and its departure time from the shaper of any visited node h is then increasing packet after packet. By Property 7, this delay is upper bounded and the bound is reached when τ_i generates packets at its maximum rate. Then, if m' is a packet of τ_i incurring this maximum delay, we get for any packet $m'+k$ of τ_i, where $k \in \mathbb{N}$: $d_{m'+k}^{shap(h)} - a_{m'+k}^1 = d_{m'}^{shap(h)} - a_{m'}^1$. Hence, $d_{m'+k}^{shap(h)} - d_{m'}^{shap(h)} = a_{m'+k}^1 - a_{m'}^1 = k \cdot T_i$. ■

Property 7 *Let τ_i be a flow belonging to the* EF *class and following line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes applying the token bucket technique to reshape flows belonging to the* EF *class. If, on each visited node, the parameters of the token bucket associated to τ_i are $(L_i, L_i/T_i)$, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max} \\ \\ Jout_i^{\mathcal{L}} = \sum_{h=1}^{q} (Rmax_i^{sched(h)} - C_i^h) + (q-1) \cdot (P_{max} - P_{min}). \end{cases}$$

Proof: By recursion. Let τ_i be a flow belonging to the EF class and following line $\mathcal{L}$. We assume that $\mathcal{L}$ consists of q nodes using token buckets $(L_i, L_i/T_i)$ to reshape τ_i. As there is no jitter on any source node, the worst case response time of τ_i in its source node is equal to the maximum sojourn time experienced by any packet of τ_i in the node scheduler. Hence, we get: $Rmax_i^1 = Rmax_i^{sched(1)}$.

We now assume that the worst case response time of τ_i when visiting $q-1$ nodes is equal to: $\sum_{h=1}^{q-1} Rmax_i^{sched(h)} + (q-2) \cdot P_{max}$. If all EF flows generate packets at their maximum rates and τ_i visits q nodes, then the response time of any packet $m \geq m'$ of τ_i, where $S_{m'}^h = Smax_i^h$, is: $R_m^{\mathcal{L}} = S_m^q + R_m^{shap(q)} + R_m^{sched(q)}$. By Property 4, we get: $R_m^{\mathcal{L}} = S_m^q + Smax_i^q - S_m^q + R_m^{sched(q)}$, that is equal to: $\sum_{h=1}^{q-1} Rmax_i^{sched(h)} + (q-2) \cdot P_{max} + P_{max} + R_m^{sched(q)}$. Hence, we have: $R_m^{\mathcal{L}} = \sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max}$.

Concerning the jitter of flow τ_i, if we consider in any node h only packets of τ_i arrived after the one having experienced the maximum delay from its arrival time in the DiffServ domain to its arrival time in node h, τ_i leaves the node shaper considered without jitter (see Section 5.3). But this is wrong if we consider all the packets of τ_i. For example, the first packet of τ_i visits all the nodes of line $\mathcal{L}$ without being delayed by any shaper. Indeed, the shaper associated to τ_i in any of the q visited nodes consists of a token bucket $(L_i, L_i/T_i)$. Then, there is enough tokens in each bucket to let the first packet of τ_i pass through all shapers. Moreover, if this packet is not delayed in any scheduler by packets of other flows, its minimum end-to-end response time is equal to: $\sum_{h=1}^{q} C_i^h + (q-1) \cdot P_{min}$.

Hence, the worst case end-to-end jitter of the EF flow τ_i, that is $Jout_i^{\mathcal{L}} = Rmax_i^{\mathcal{L}} - Rmin_i^{\mathcal{L}}$, meets: $Jout_i^{\mathcal{L}} \leq \sum_{h=1}^{q} (Rmax_i^{sched(h)} - C_i^h) + (q-1) \cdot (P_{max} - P_{min})$. This bound is reached when flow τ_i generates packets at its maximum rate. ■

Property 8 *Let τ_i be a flow belonging to the* EF *class. Let $\mathcal{L}$ be a line followed by τ_i, it consists of q nodes numbered from 1 to q. If the scheduling of the* EF *class is* FIFO, *then the delay due to the non-preemption is bounded by: $\sum_{h=1}^{q} (B^h - 1)$, where B^h denotes the maximum processing time at node h of any packet of flow not belonging to the* EF *class.*

Proof: As detailed in Section 4, on any node h, the delay due to the non-preemption and incurred by any EF packet m consists in two parts. Indeed, m may incur a delay directly due to a packet not belonging to the EF class if it enters the node scheduler while such a packet is in processing. Moreover, m may incur an additional delay due to EF packets arriving in the scheduler of node h while m is waiting for processing and having priority over m. If the scheduling of the EF class is FIFO, no EF packet arrived after m can be scheduled before m. Hence, the worst case non-preemptive effect is limited to the maximum delay directly due to packets not belonging to the EF class. From Property 1, this maximum delay is equal to: $\sum_{h=1}^{q}(B^h - 1)$. ■

Property 9 *In the uniprocessor context, if the scheduling of the* EF *class is* FIFO *and the* EF *flows release packets without jitter, then the worst case response time of any* EF *flow τ_i, in the presence of other classes flows, is obtained in the first busy period of the synchronous scenario, that is packets of the* EF *flows are released at the same time, and is equal to: $R_{max_i} = \sum_{j=1}^{n} C_j + (B-1)$.*

Proof: An extension of a result proved in [GEO98], when EF flows coexist with flows of other classes. ■

Property 10 *In the uniprocessor context, if the scheduling of the* EF *class is* FIFO *and the* EF *flows release packets with jitter, the worst case response time of any* EF *flow τ_i, in the presence of other classes flows, is equal to:*

$$R_{max_i} = \max_{-Jin_i \leq t < L} \left\{ \sum^{n}_{\substack{j=1 \\ t \geq -Jin_j}} \left(1 + \left\lfloor \frac{t + Jin_j}{T_j} \right\rfloor\right) \cdot C_j - t \right\} + (B-1),$$

where Jin_j is the release jitter of EF *flow τ_j and L is the length of the longest busy period, first solution of $L = \sum_{j=1}^{n} \lceil (L + Jin_j)/T_j \rceil \cdot C_j$.*

Proof: An extension of a result proved in [GEO98], when EF flows coexist with flows of other classes. ■

Property 11 *Let $\tau = \{\tau_1, \tau_2, ..., \tau_n\}$ be the set of the n sporadic flows belonging to the* EF *class in the DiffServ domain $\mathcal{D}$. All the* EF *flows follow the same line $\mathcal{L}$ in $\mathcal{D}$. If the scheduling of the* EF *class is* FIFO*, then we have:*

$$\forall \tau_i \in \tau,\ R_i^{\mathcal{L}} \leq \sum_{j=1}^{n} C_j^{slow} + \sum_{\substack{h=1 \\ h \neq slow}}^{q} C_{max}^h + \sum_{h=1}^{q} (B^h - 1) + (q-1) \cdot P_{max},$$

where slow *denotes the slowest node among the q visited nodes and $C_{max}^h = \max_{j=1..n}(C_j^h)$.*

Proof: See [GMM02]. ■

Property 12 *If the condition* $\sum_{h=1}^{q} \sum_{\substack{j=1 \\ first_j=h}}^{n} (C_j^{slow_j}/T_j) \leq 1$ *is met, then the worst case end-to-end response time of any packet* m *of* EF *flow* τ_i *is bounded by:*

$$Rmax_i^{\mathcal{L}} \leq \sum_{h=1}^{q} \left(\sum_{\substack{j=1 \\ first_j=h}}^{n} \left(1 + \frac{S_m^h + Jin_j^h}{T_j}\right) \cdot C_j^{slow_j} \right) + \sum_{\substack{h=1 \\ h \neq slow}}^{q} C_{max}^h + \sum_{h=1}^{q} (B^h - 1) + (q-1) \cdot P_{max},$$

Proof: See [MMG03]. ■

Property 13 *Let* $\mathcal{D}$ *be a DiffServ domain where all nodes reshape flows belonging to the* EF *class applying the jitter cancellation technique. If the scheduling of the* EF *class is* FIFO*, then the worst case end-to-end response time of any* EF *flow is obtained considering the synchronous scenario in each visited node.*

Proof: As the jitter cancellation technique is used in all the nodes of the DiffServ domain to reshape flows belonging to the EF class, any EF flow τ_i enters the scheduler of any node without release jitter, since its packets are hold in the shaper until their latest possible arrival times. Hence, if τ_i visits nodes h and $h+1$, the arrival times of its packets on the scheduler of node h are all translated with the same value on the scheduler of node $h+1$.

According to Property 9, all the EF packets visiting node h experience the same worst case response time in the scheduler of this node due to the scheduling of the EF class, that is FIFO. This worst case response time is obtained when all the EF flows are synchronous on the node scheduler. Hence, if the EF flows visiting nodes h and $h+1$ are synchronous on node h, they will be synchronous on node $h+1$. ■

Property 14 *Let* τ_i *be a flow belonging to the* EF *class and following line* $\mathcal{L}$*, where* $\mathcal{L}$ *consists of* q *nodes applying the jitter cancellation technique to reshape flows belonging to the* EF *class. If the* EF *class is scheduled* FIFO*, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} = \sum_{h=1}^{q} \sum_{j=1}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot P_{max} \\ Jout_i^{\mathcal{L}} \;\; = \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^q + (B^q - 1). \end{cases}$$

Proof: The worst case end-to-end response time of any EF flow τ_i following a line $\mathcal{L}$ consisting of q nodes is equal to: $\sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max}$ (see Property 2). Moreover, if the scheduling of the EF class is FIFO, the worst case end-to-end response time of any EF flow is obtained considering the synchronous scenario in each visited node (see Property 13). As flows belonging to the EF class enter the scheduler of any node without release jitter, we get: $Rmax_i^{sched(h)} = \sum_{j=1}^{n} C_j^h + \left(B^h - 1\right)$, for any visited node h (see Property 9). ■

Property 15 *Let τ_i be a flow belonging to the* EF *class reshaped by a token bucket $(L_i, L_i/T_i)$ on each visited node. If the scheduling of the* EF *class is* FIFO*, then a bound on the worst case end-to-end response time of τ_i is obtained considering the synchronous scenario in each visited node.*

Proof: For any EF flow τ_i, for any node h visited by τ_i, the token bucket $(L_i, L_i/T_i)$ guarantees T_i as the minimum inter-arrival time on the scheduler of node h between two packets of τ_i. Hence, from the node scheduler point of view, EF flows are released without jitter. Moreover, since the scheduling of the EF class is FIFO, the worst case response time of any EF flow in the scheduler of node h is obtained in the synchronous scenario (see Property 9) when all the EF flows generate packets at their maximum rates.

Let us assume that on any node h of the line $\mathcal{L}$ followed by τ_i, the EF flows generate packets at their maximum rates and experience in turn the worst case response time in the node scheduler, then any EF flow has experienced the worst case response time at the latest after $n \cdot \text{LCM}_{j=1..n}(T_j)$, where n denotes the number of EF flows in the DiffServ domain. Hence, if the EF flows visiting nodes h and $h+1$ are synchronous on node h, they will be synchronous on node $h+1$ at the latest after $n \cdot \text{LCM}_{j=1..n}(T_j)$. ∎

Property 16 *Let τ_i be a flow belonging to the* EF *class reshaped by a token bucket $(L_i, L_i/T_i)$ on each visited node. Let $\mathcal{L}$ be a line followed by τ_i, where $\mathcal{L}$ consists of q nodes. If the scheduling of the* EF *class is* FIFO*, then:*

$$\begin{cases} Rmax_i^{\mathcal{L}} \leq \sum_{h=1}^{q} \sum_{j=1}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot P_{max} \\ Jout_i^{\mathcal{L}} \leq \sum_{h=1}^{q} \sum_{\substack{j=1 \\ j \neq i}}^{n} C_j^h + \sum_{h=1}^{q} \left(B^h - 1\right) + (q-1) \cdot (P_{max} - P_{min}), \end{cases}$$

Proof: The worst case end-to-end response time of any EF flow τ_i following a line $\mathcal{L}$, where $\mathcal{L}$ consists of q nodes, is equal to: $\sum_{h=1}^{q} Rmax_i^{sched(h)} + (q-1) \cdot P_{max}$ (see Property 7). By Properties 9 and 15, if the EF class is scheduled FIFO, we get: $\forall\, h \in [1, q]$, $Rmax_i^{sched(h)} \leq \sum_{j=1}^{n} C_j^h + \left(B^h - 1\right)$. ∎

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399